

22nd March 2019

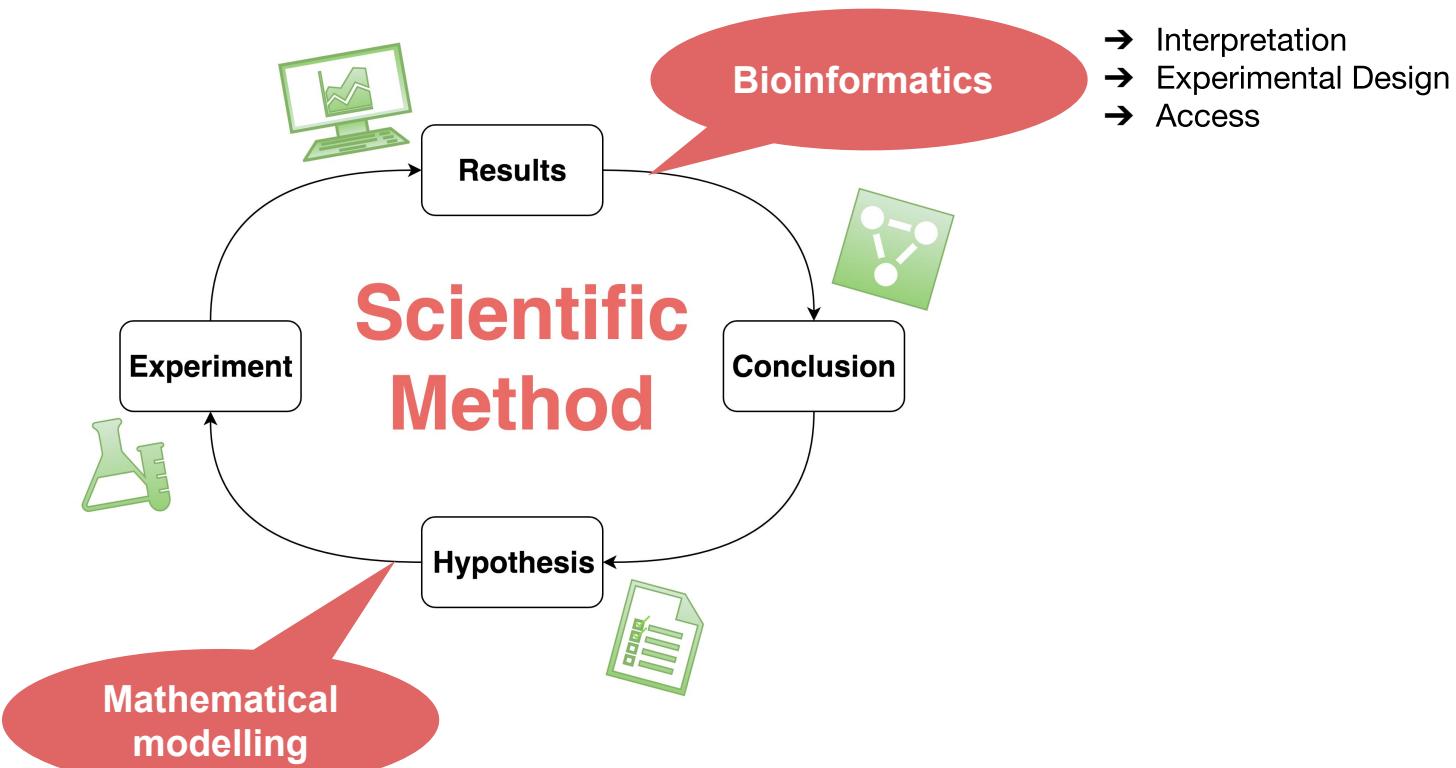
International Spring School “Computational Biology Starter”, IPK Gatersleben

Metabolic Modelling

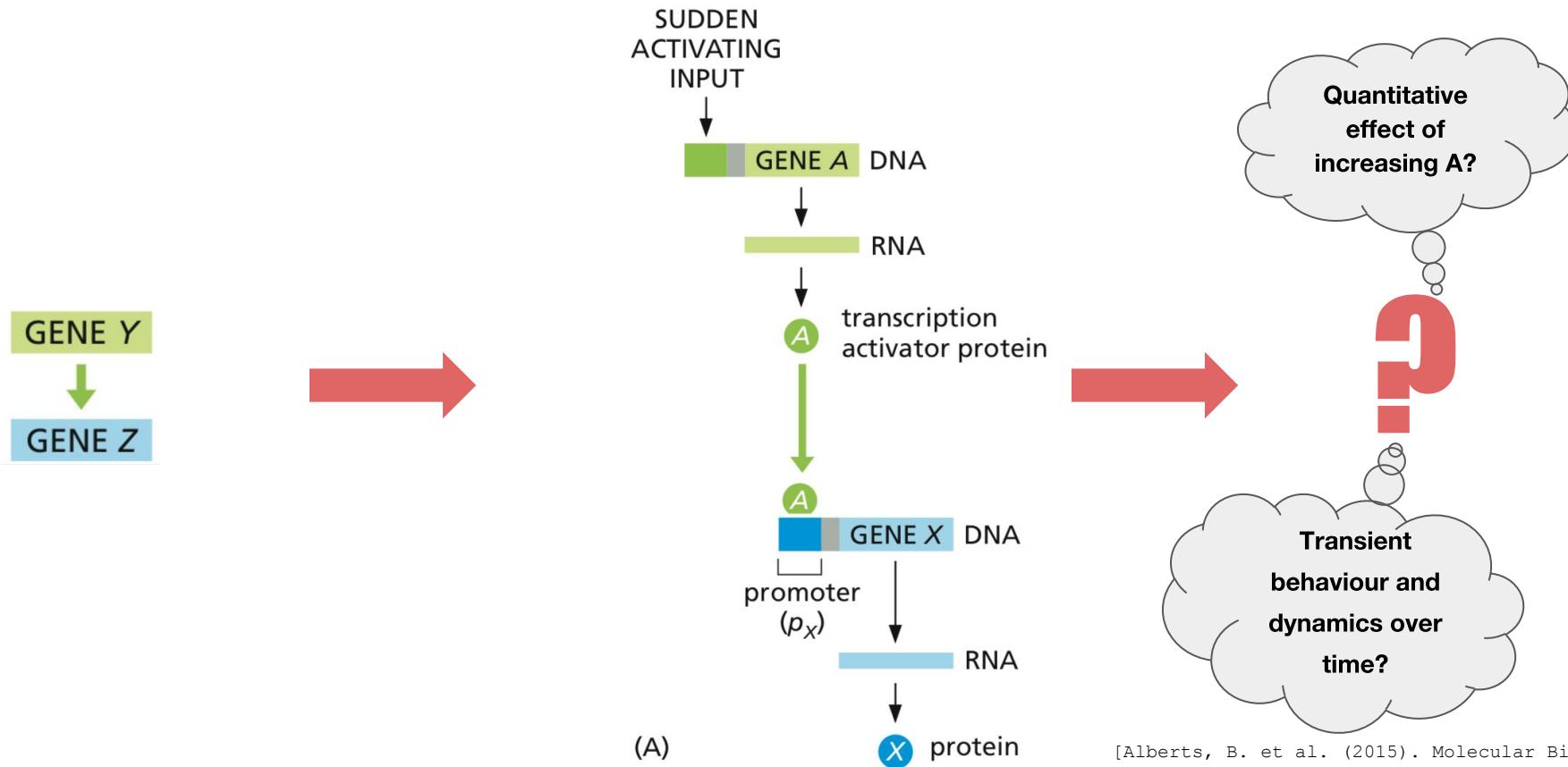
A Crash Course

Mary-Ann Blätke, Jędrzej Szymański

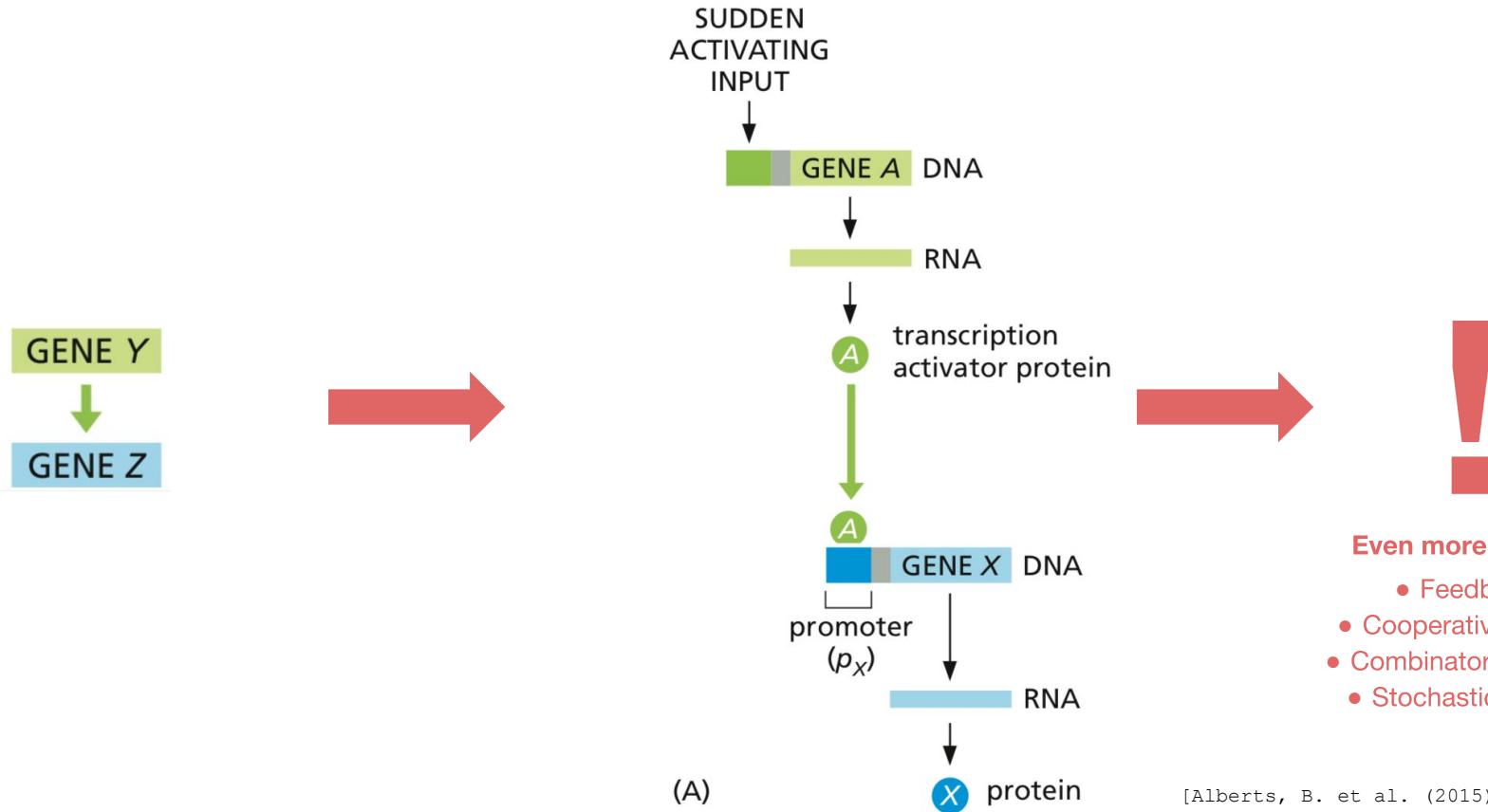
Bioinformatics vs. Mathematical Modelling



Human Guessing vs. Mathematical Reasoning



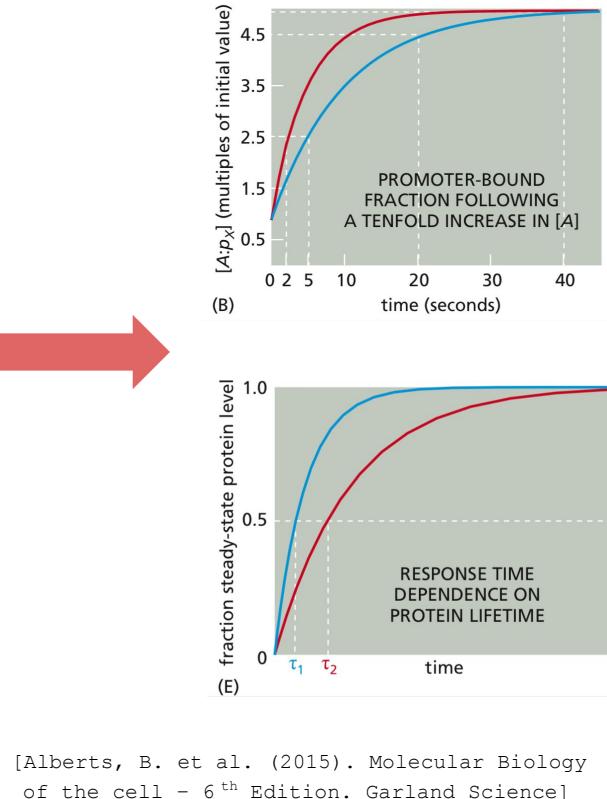
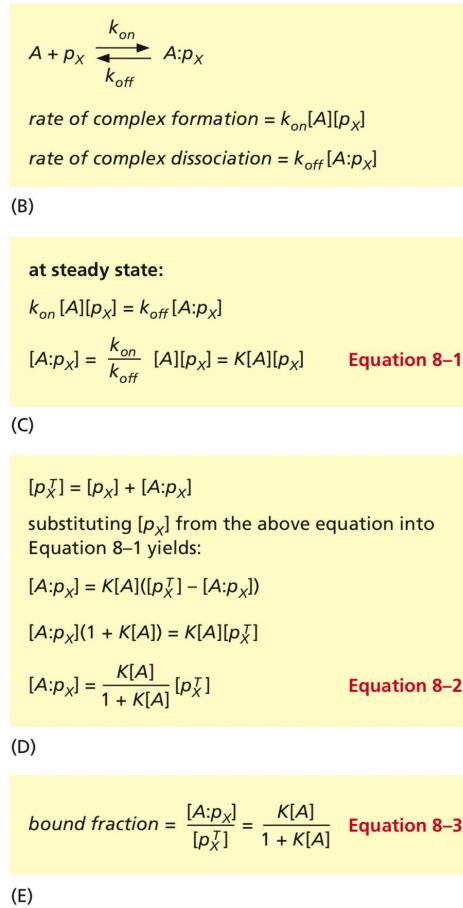
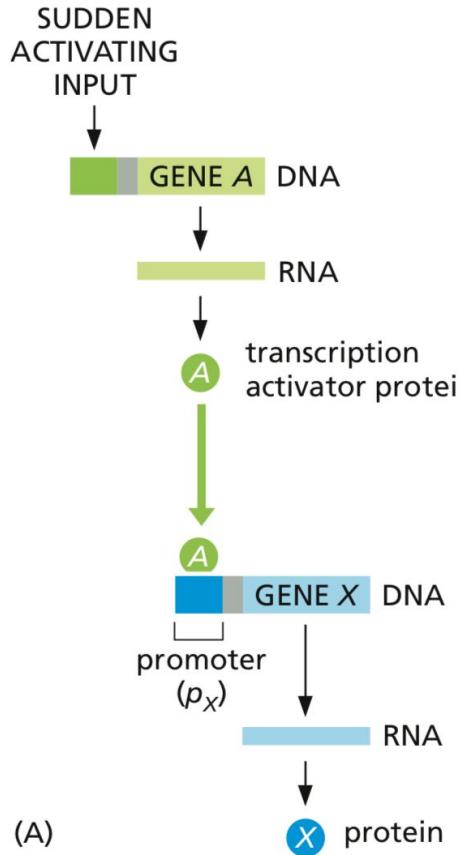
Human Guessing vs. Mathematical Reasoning



Even more difficult:

- Feedback
- Cooperative effects
- Combinatorial control
- Stochastic effects

Human Guessing vs. Mathematical Reasoning



Power of Mathematical Modelling

**Using accurate logic to formally express and integrate
biological information**

enables:

Rigorous testing of biological hypotheses

“If mathematical reasoning from a given hypothesis leads to a prediction that is not true, then the hypothesis is not true.”

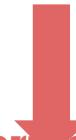
Power of Mathematical Modelling

Advantages

- Easy to build and modify
- Coherent and simplified description of large and complex real-world systems
- Idea of the important in-/outputs
- Improvement of understanding
- Probing robustness
- Prediction of the systems behaviour
- Controllability of the real-world systems

Disadvantages

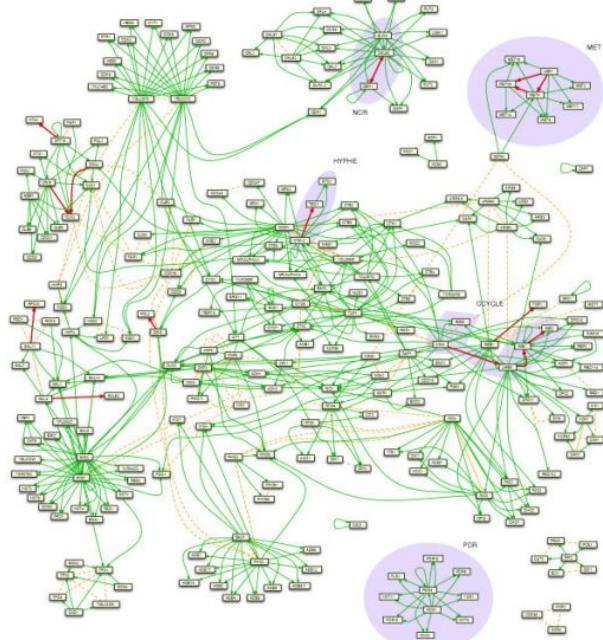
- Oversimplification
- Valid only under assumed conditions


**Experimental Verification
Needed!**

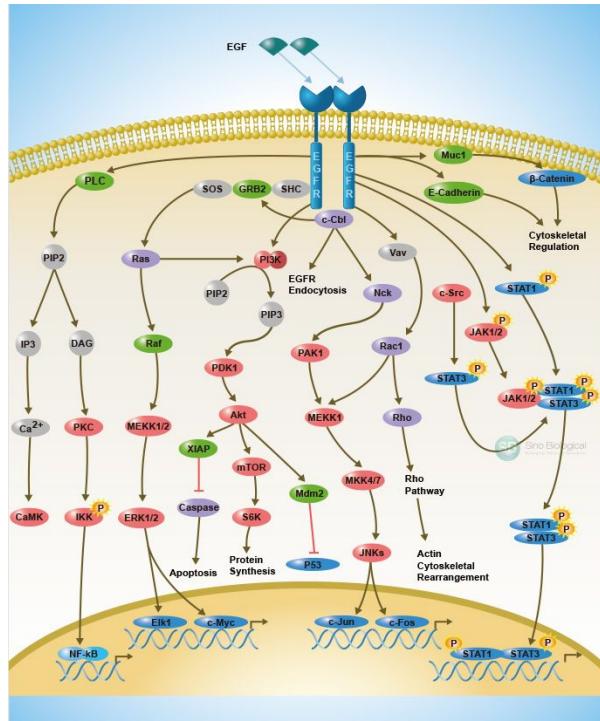
Still

Intracellular Network Types

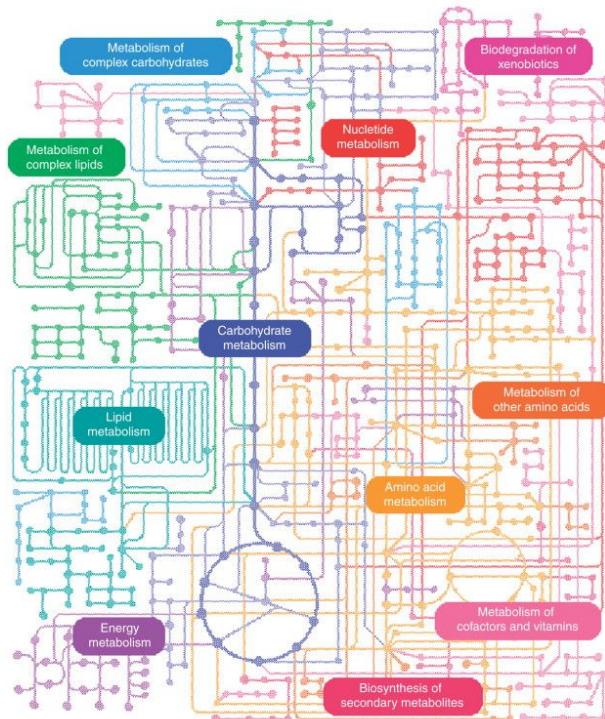
Gene Regulatory Networks



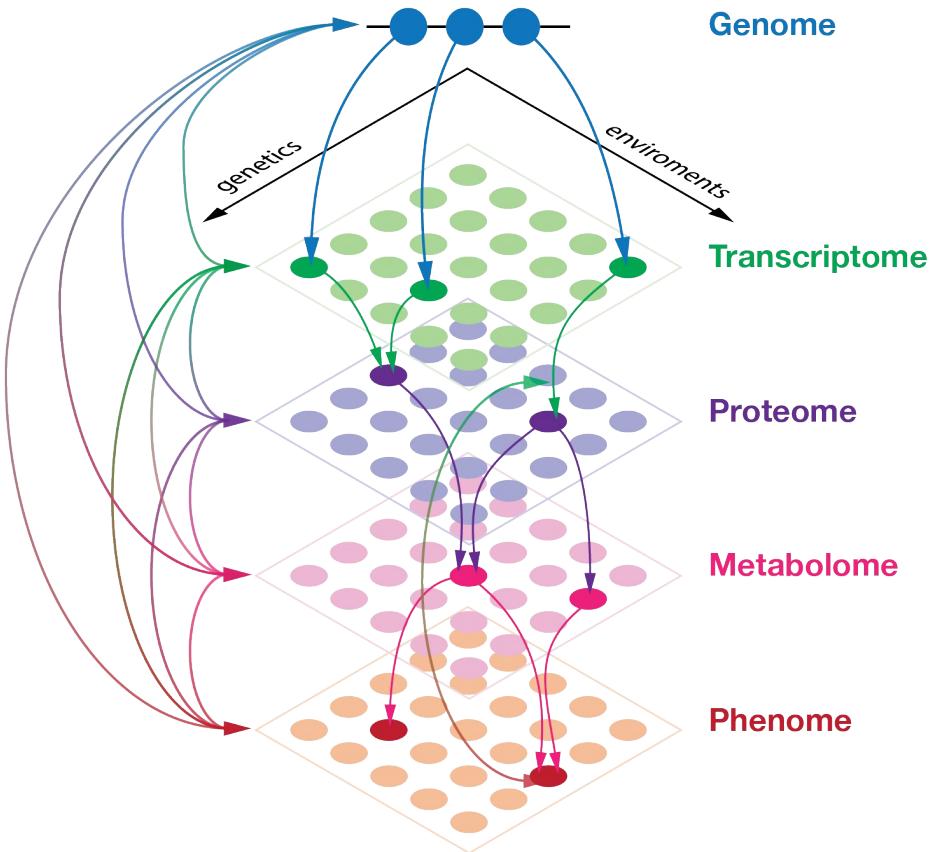
Signaling Networks



Metabolic Networks



Intracellular Network Types

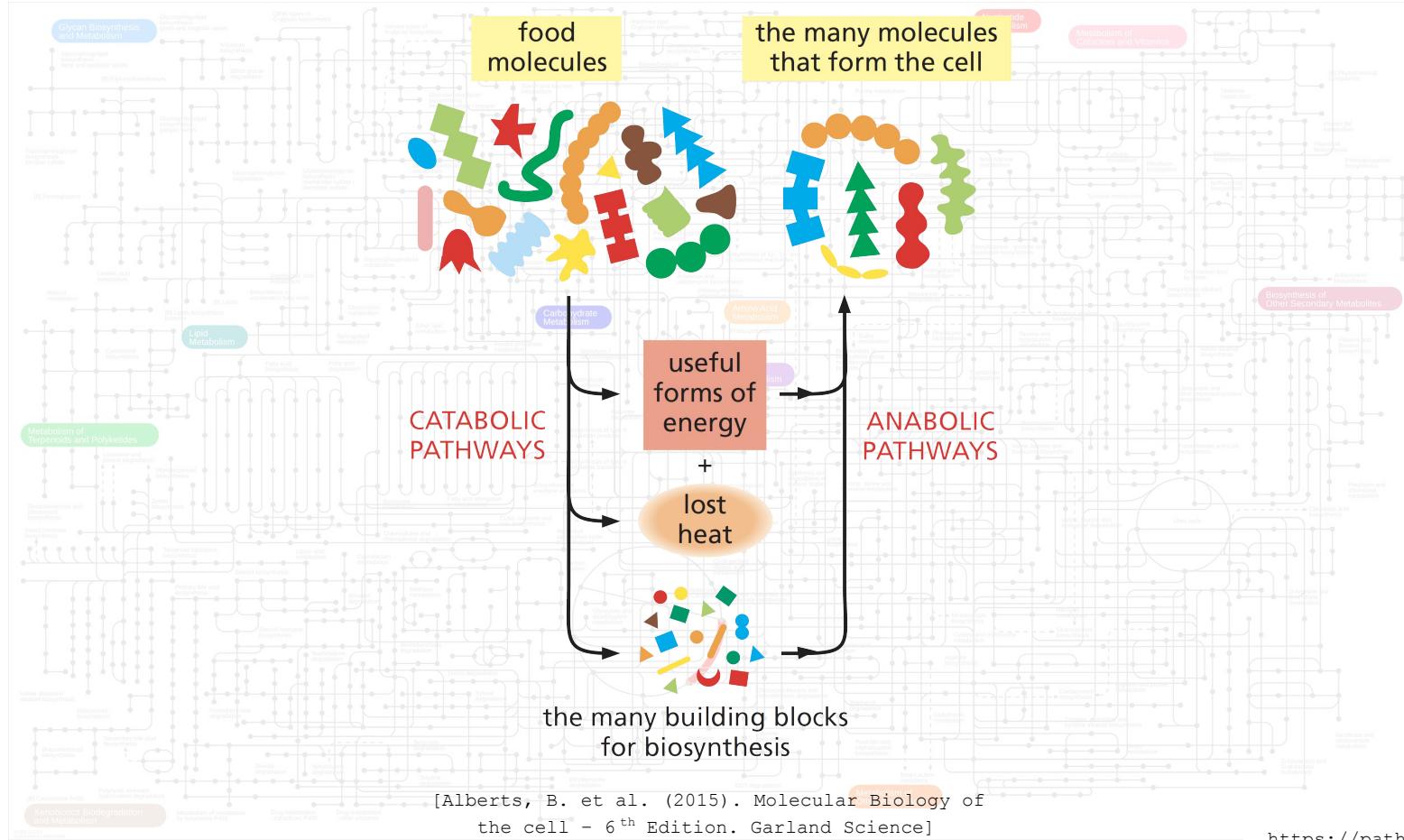


- Networks of different types are highly interconnected
- They involve large variety of molecules and interactions
- Different modelling & analysis methods are required depending on the network type, size and available information, precise question

HERE WE WILL COVER:
Metabolic Networks
Constraint-based Modelling/Analysis

&

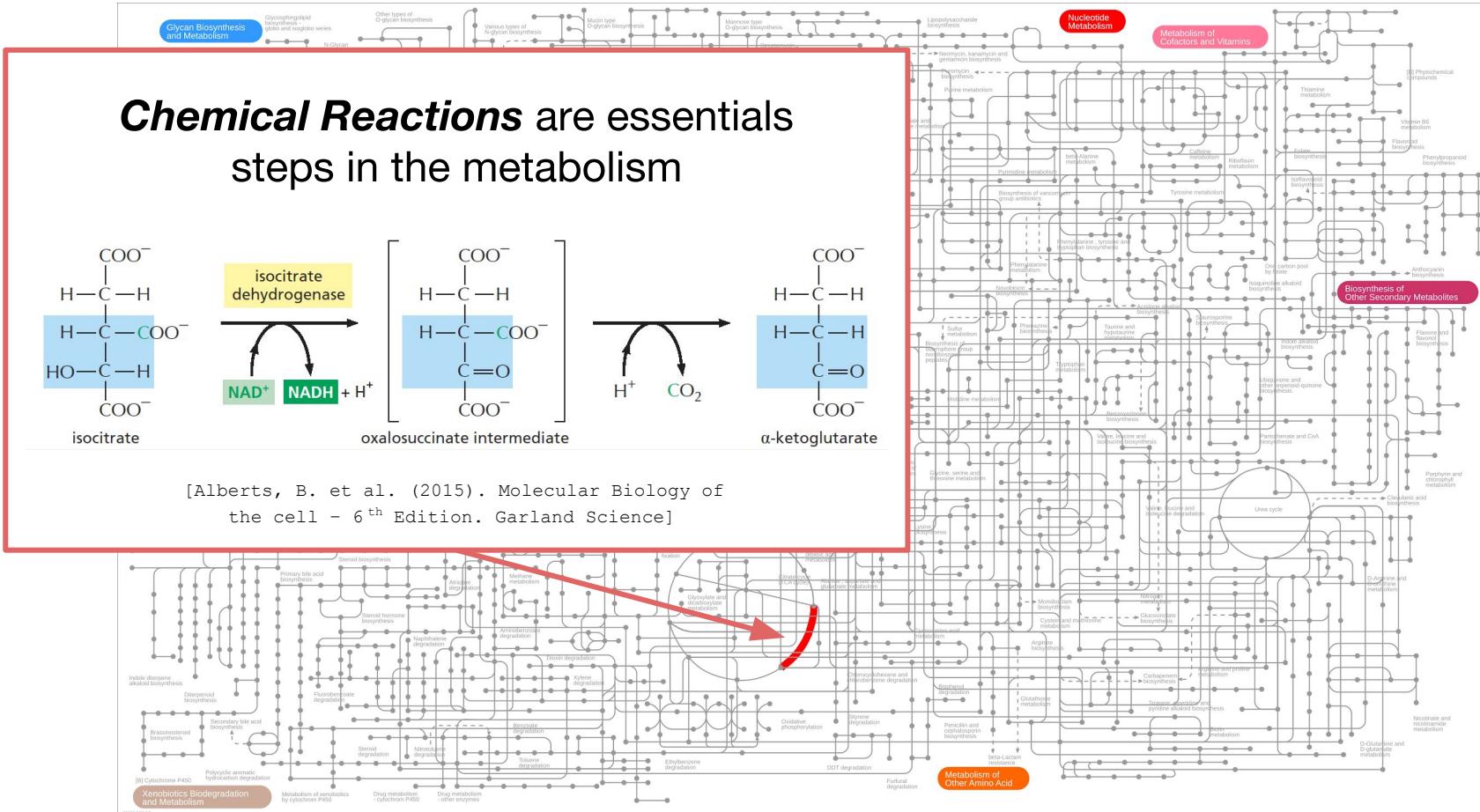
Metabolic Networks



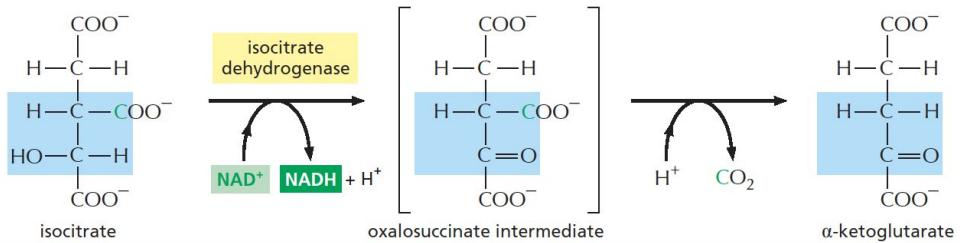
[Alberts, B. et al. (2015). Molecular Biology of the cell - 6th Edition. Garland Science]

<https://pathways.embl.de>

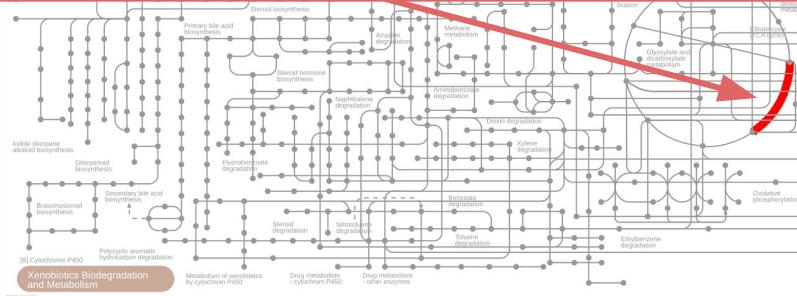
Metabolic Networks - Chemical Reactions



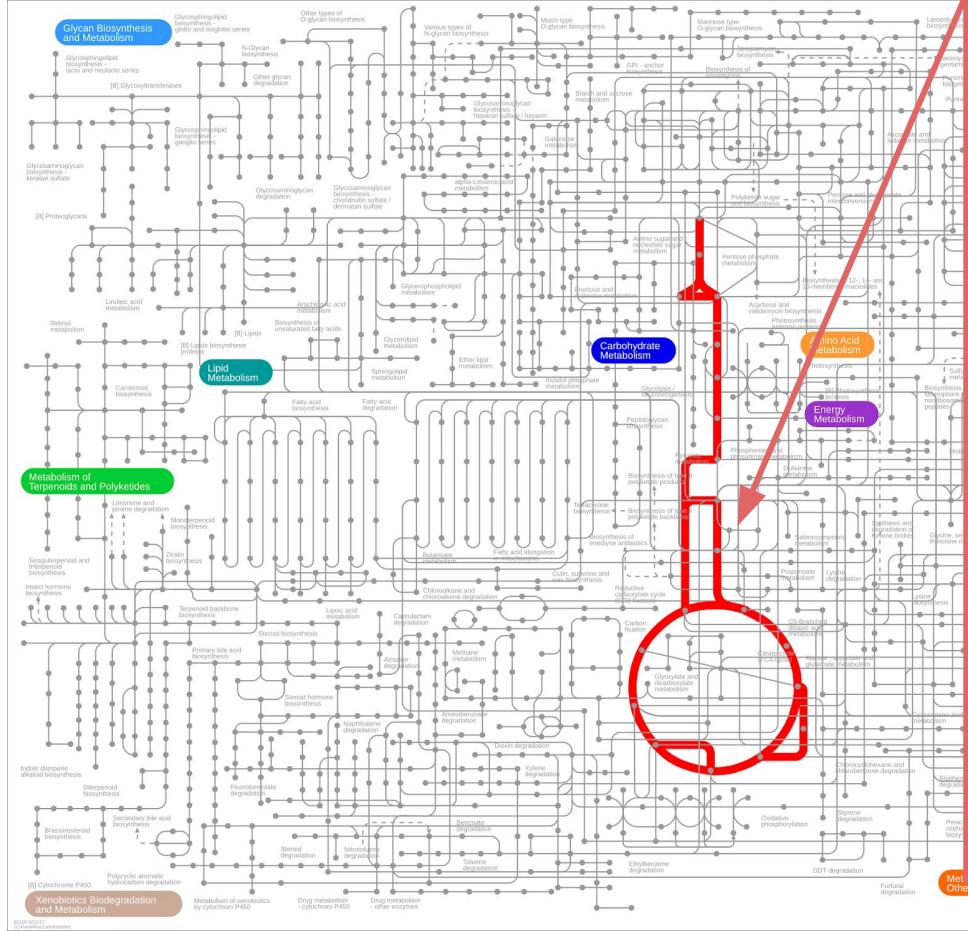
Chemical Reactions are essentials steps in the metabolism



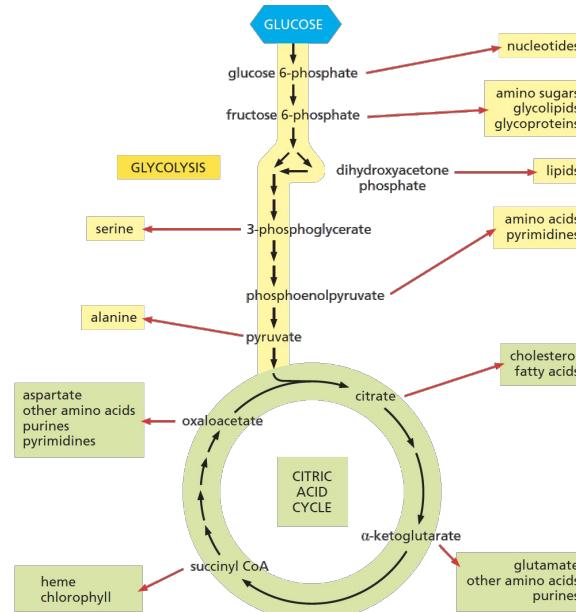
[Alberts, B. et al. (2015). Molecular Biology of the cell - 6th Edition. Garland Science]



Metabolic Networks - Pathways



Pathways are building blocks of related chemical reactions

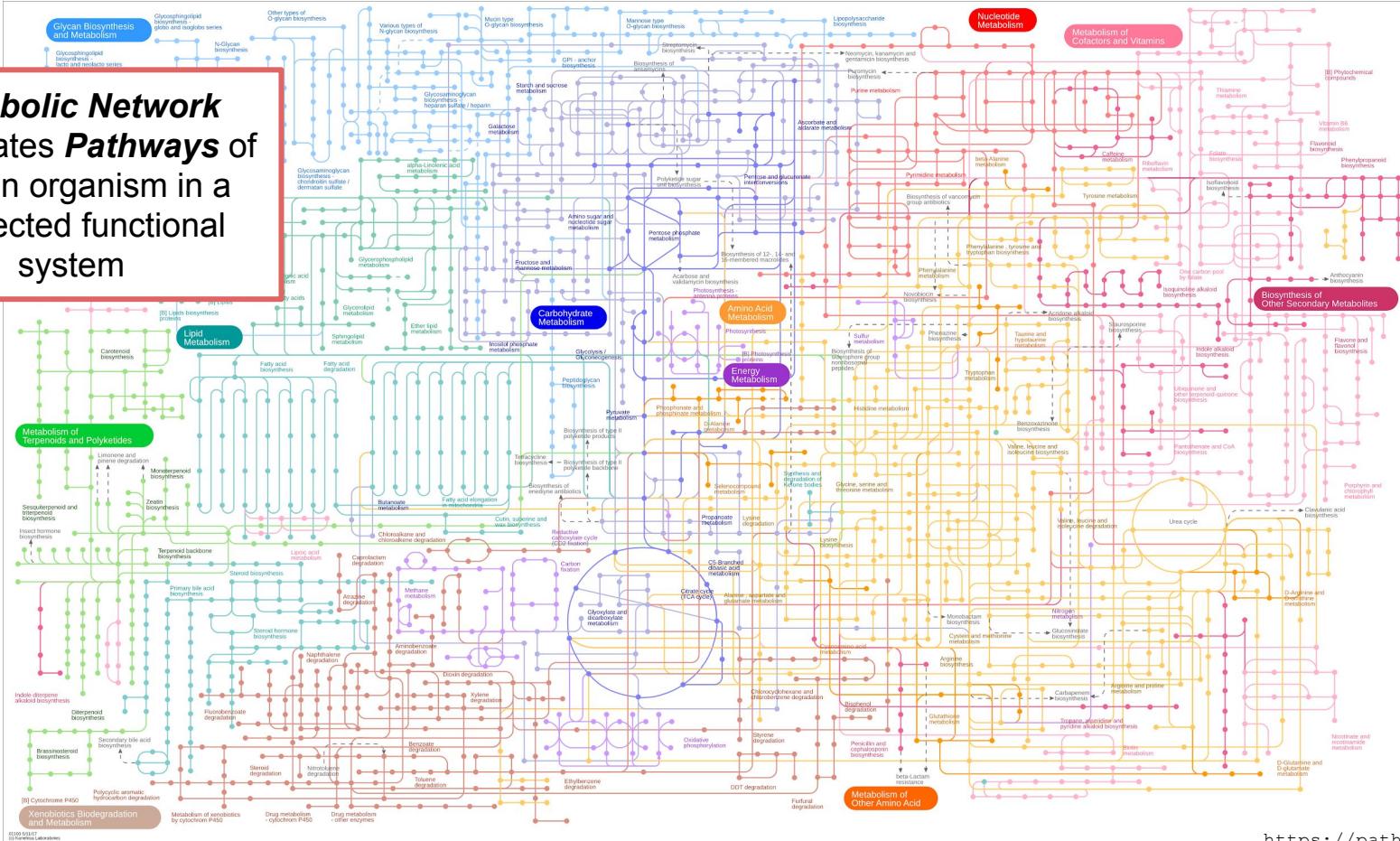


Glycolysis Citric Acid Cycle

[Alberts, B. et al. (2015). Molecular Biology of the cell - 6th Edition. Garland Science]

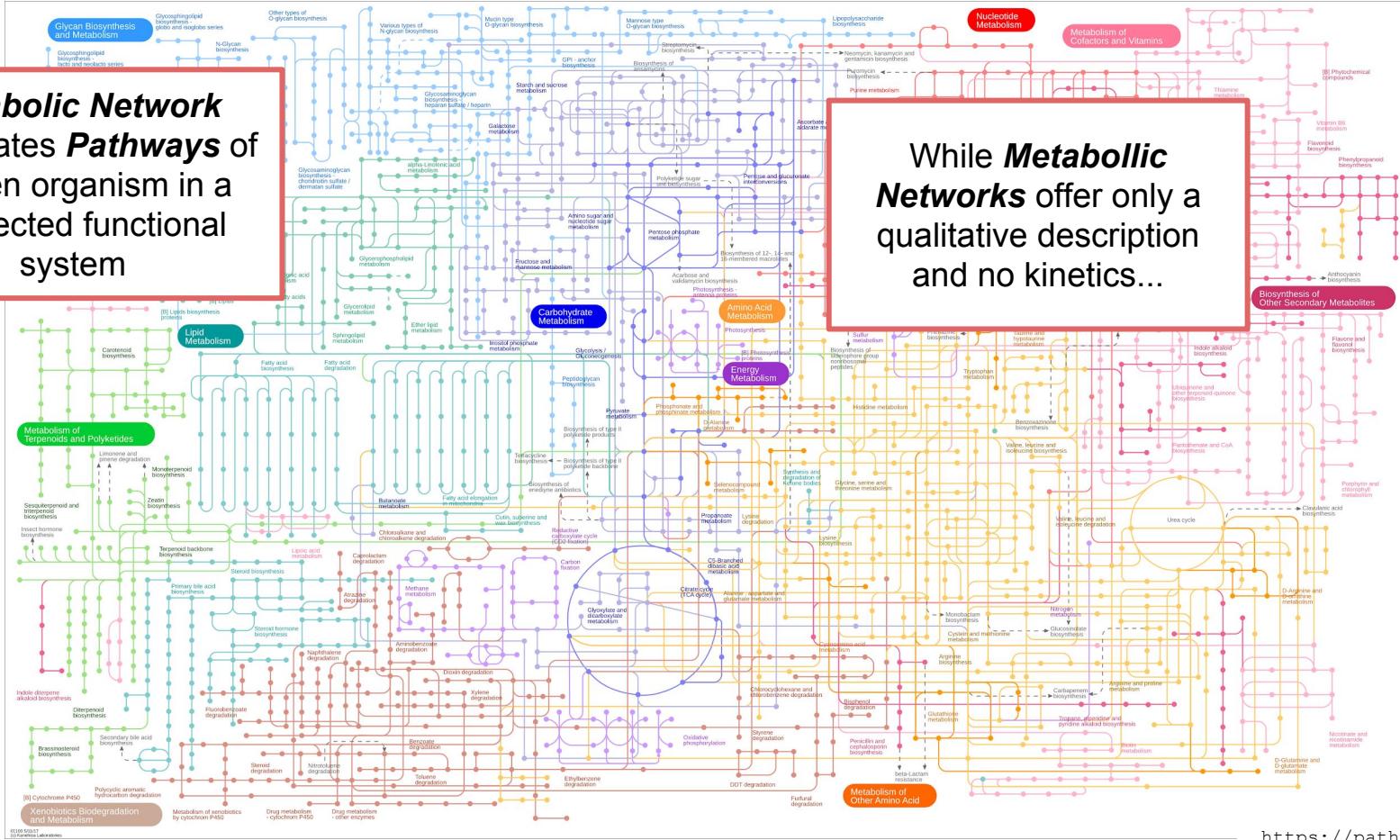
Metabolic Networks

Metabolic Network
aggregates **Pathways** of
a given organism in a
connected functional
system



Metabolic Networks

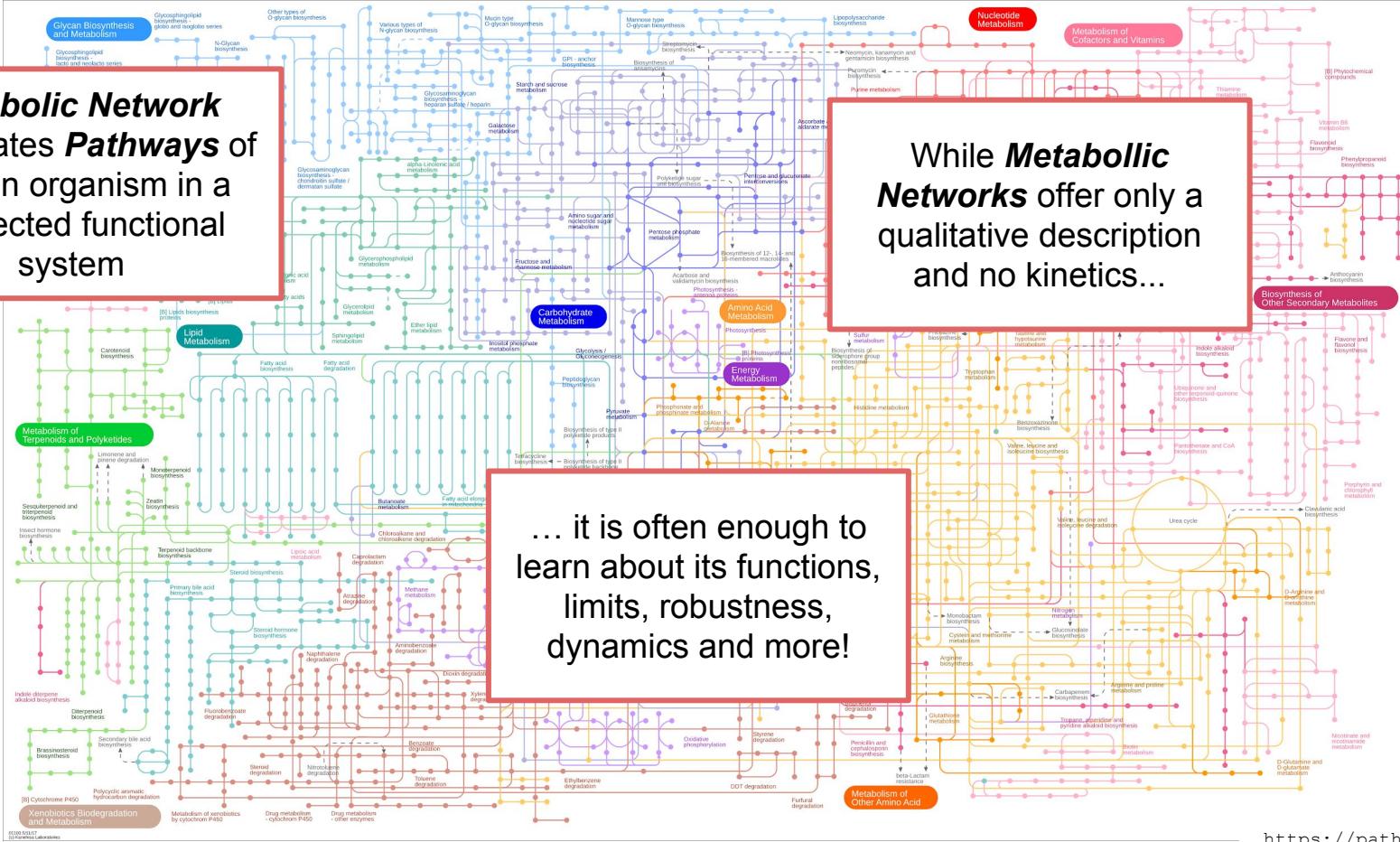
Metabolic Network
aggregates **Pathways** of
a given organism in a
connected functional
system



While ***Metabolic Networks*** offer only a qualitative description and no kinetics...

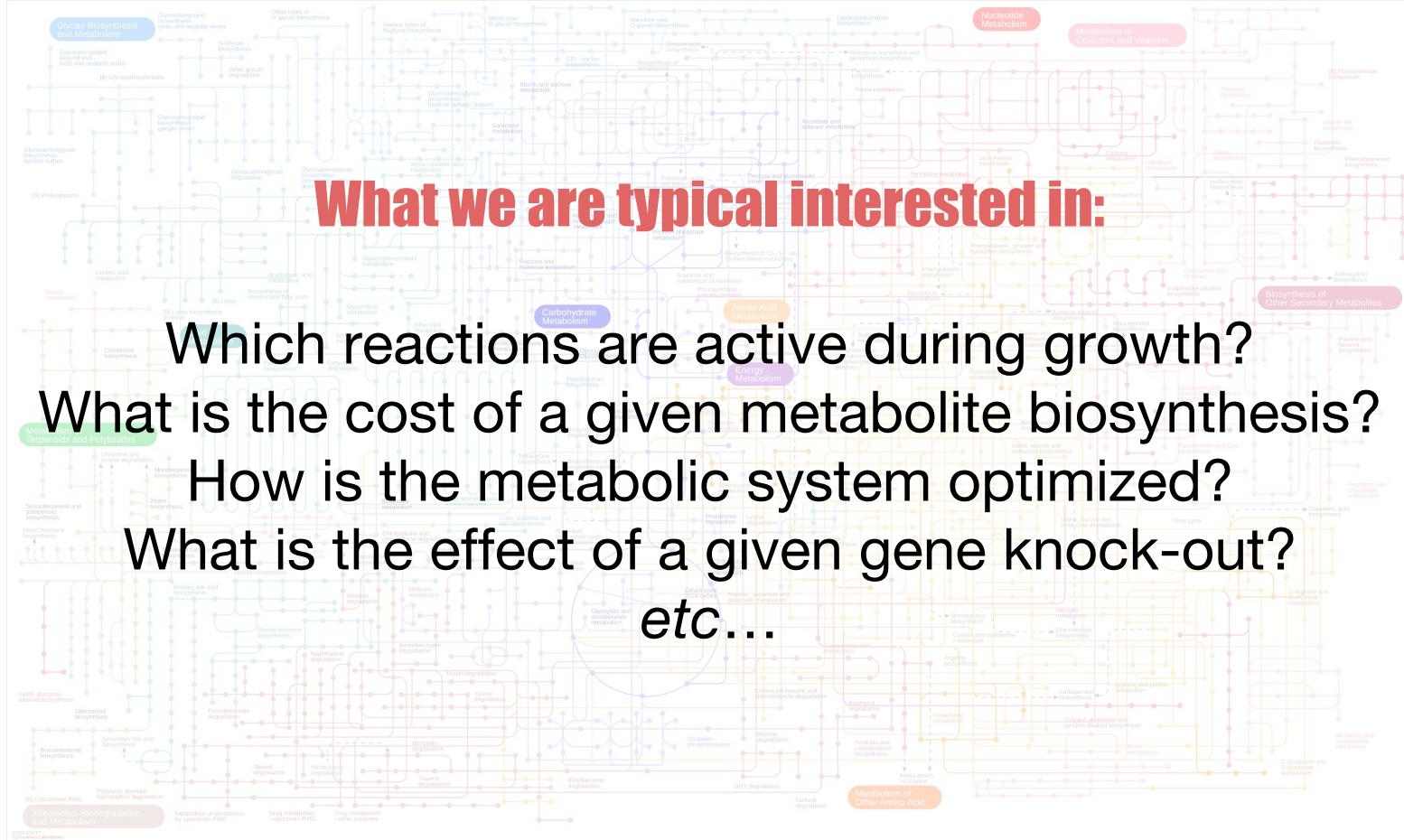
Metabolic Networks

Metabolic Network
aggregates **Pathways** of
a given organism in a
connected functional
system



... it is often enough to
learn about its functions,
limits, robustness,
dynamics and more!

Metabolic Network



What we are typical interested in:

Which reactions are active during growth?

What is the cost of a given metabolite biosynthesis?

How is the metabolic system optimized?

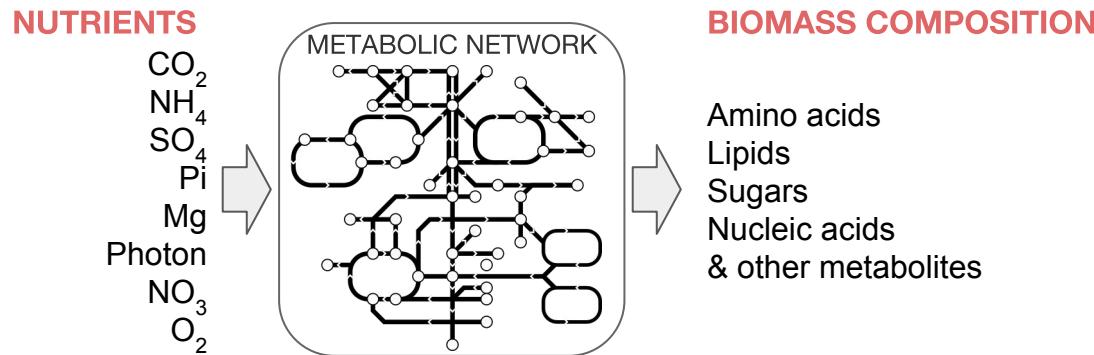
What is the effect of a given gene knock-out?

etc...

Flux Balance Analysis

Makes use of:

- Metabolic network → primary & secondary metabolism of interest
- Set of constraints
 - Limitations on inputs and outputs → nutrients & biomass/product
 - Thermodynamic constraints → directionality of reactions
 - Enzyme capacity → flux boundaries
- Objective function → metabolic function to be optimized (e.g. maximization of growth)

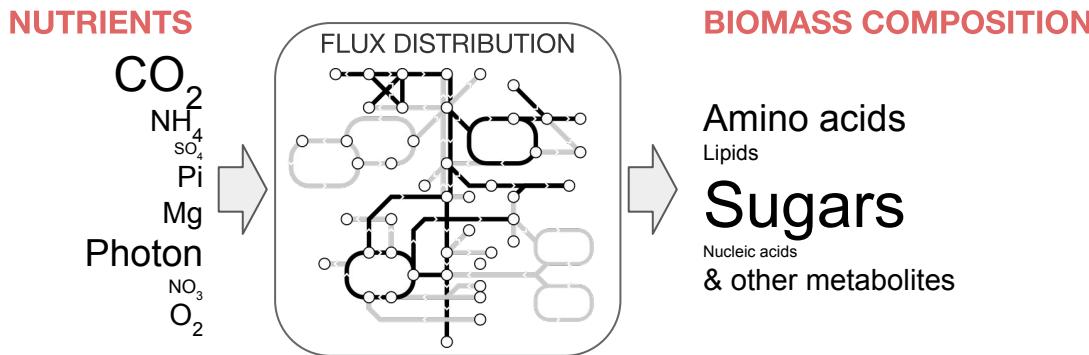


Flux Balance Analysis

Makes use of:

- Metabolic network → primary & secondary metabolism of interest
- Set of constraints
 - Limitations on inputs and outputs → nutrients & biomass/product
 - Thermodynamic constraints → directionality of reactions
 - Enzyme capacity → flux boundaries
- Objective function → metabolic function to be optimized (e.g. maximization of growth)

To identify metabolic routes active under specific conditions



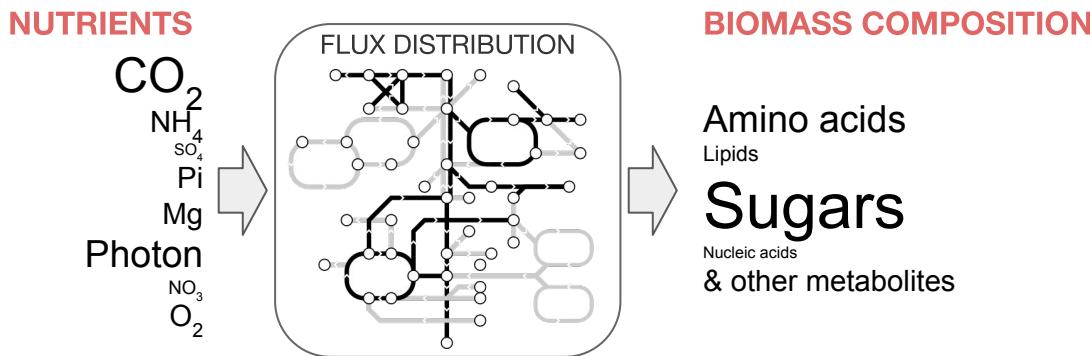
Flux Balance Analysis - Constraint-based Modelling

Makes use of:

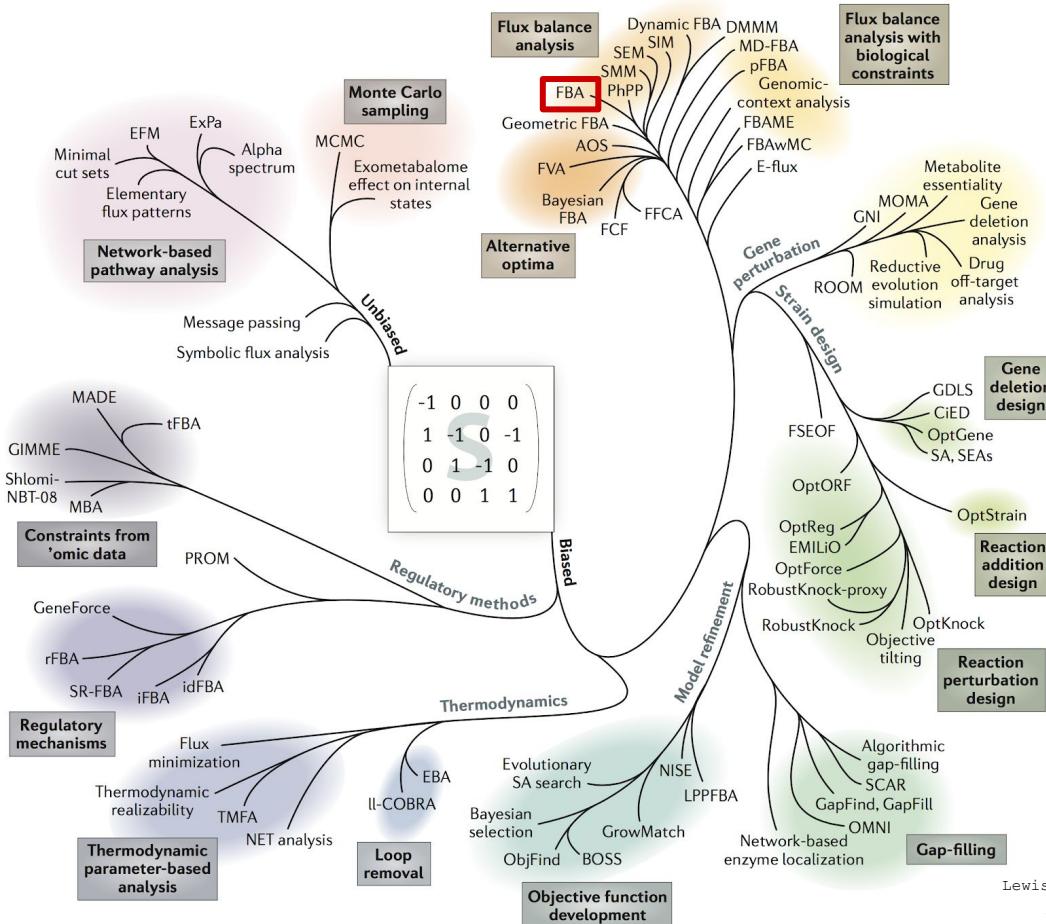
- Metabolic network → primary & secondary metabolism of interest
- Set of constraints
 - Limitations on inputs and outputs → nutrients & biomass/product
 - Thermodynamic constraints → directionality of reactions
 - Enzyme capacity → flux boundaries
- Objective function → metabolic function to be optimized (e.g. maximization of growth)



To identify metabolic routes active under specific conditions



Phylogenetic Tree of Constraint-based Modelling



Flux Balance Analysis

1 Stoichiometric Matrix

Metabolites

$$\begin{matrix} & \text{Reaction} \\ \begin{matrix} A \\ B \\ C \end{matrix} & \left(\begin{matrix} r_1 & r_2 & r_3 \\ -1 & -1 & 0 \\ 1 & 0 & -1 \\ 0 & 1 & 1 \end{matrix} \right) \end{matrix} = S$$

2 Mass Balance

$$\frac{d\vec{x}}{dt} = S * \vec{v}$$

$\vec{x} = \begin{pmatrix} x_A \\ x_B \\ x_c \end{pmatrix}$ – Concentration Vector

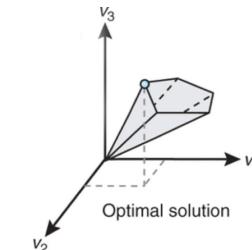
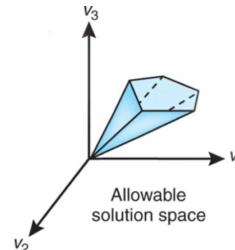
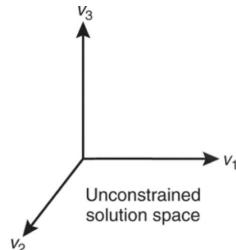
$$\vec{v} = \begin{pmatrix} v_1 \\ v_2 \\ v_3 \end{pmatrix}$$
 – Flux Vector

3 Constraints

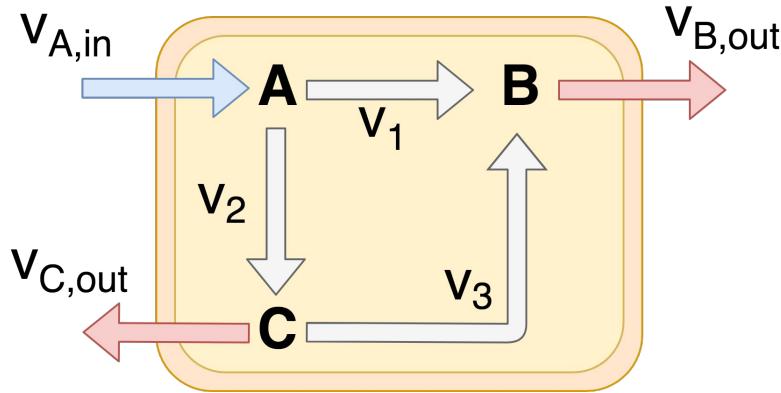
- Steady State $S * V = 0$
- Thermodynamics $v_{i,min} \leq v_i \leq v_{i,max}$
- Enzyme capacities
- Limited nutrient uptake
- Reaction ratios

4 Objective

- Maximize output of metabolite(s)
- Maximize Biomass
- Minimize use of energy
- Minimize use of nutrients
- Minimize Total Flux



Example



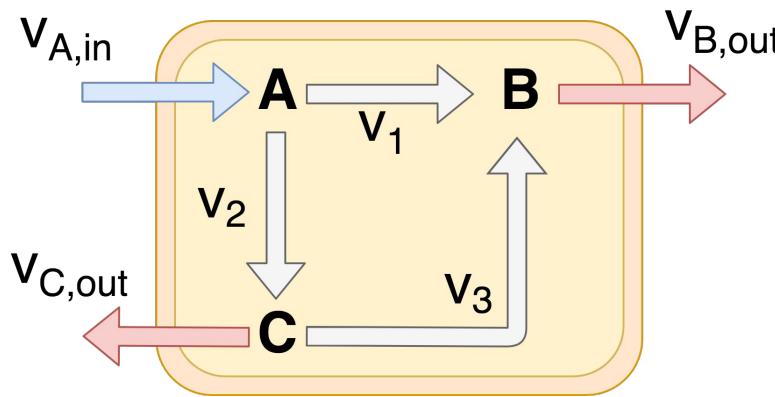
ODE's:

$$\frac{dA}{dt} = v_{A,in} - v_1 - v_2$$

$$\frac{dB}{dt} = v_1 + v_3 - v_{B,out}$$

$$\frac{dC}{dt} = v_2 - v_3 - v_{C,out}$$

Example



Constraints:

$$0 \leq v_1 \leq \infty$$

$$0 \leq v_{A,in} \leq 1$$

$$0 \leq v_2 \leq \infty$$

$$0 \leq v_{B,out} \leq 1$$

$$0 \leq v_3 \leq \infty$$

$$v_{C,out} = 0$$

Steady State Assumption:

$$v_{A,in} = v_1 + v_2$$

$$v_{B,out} = v_1 + v_3$$

$$v_{C,out} = v_2 - v_3$$

$$0 \leq v_1 + v_2 \leq 1$$

$$0 \leq v_1 + v_3 \leq 1$$

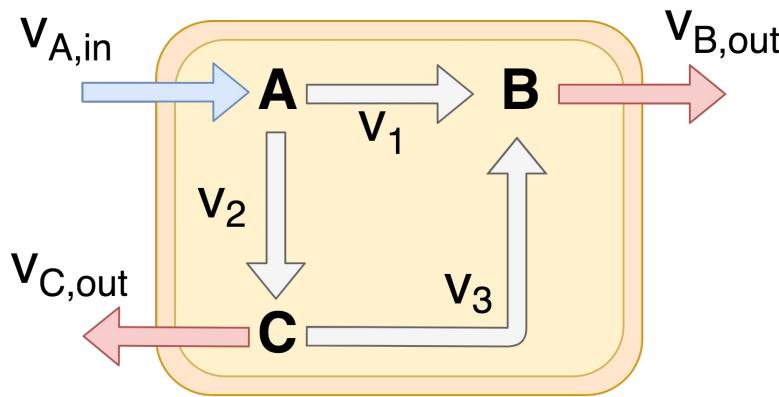
$$0 = v_2 - v_3$$

$$v_2 = v_3$$

Objective:

Maximize $v_{B,out}$

Example



Constraints:

$$0 \leq v_1 \leq \infty \quad 0 \leq v_{A,in} \leq 1$$

$$0 \leq v_2 \leq \infty \quad 0 \leq v_{B,out} \leq 1$$

$$0 \leq v_3 \leq \infty \quad v_{C,out} = 0$$

Objective:

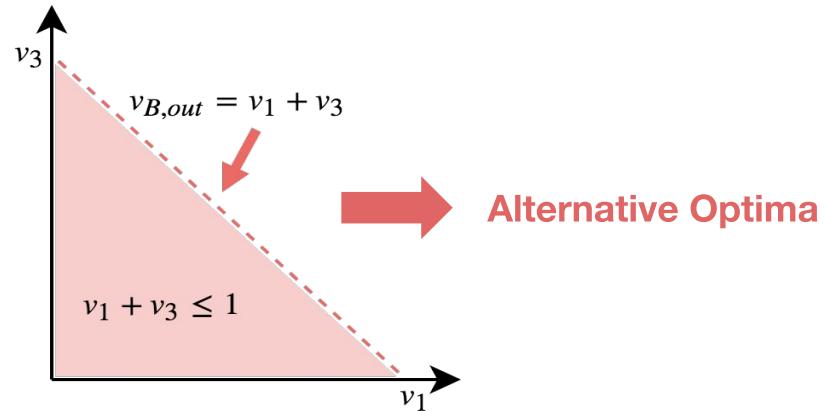
Maximize $v_{B,out}$

Steady State Assumption:

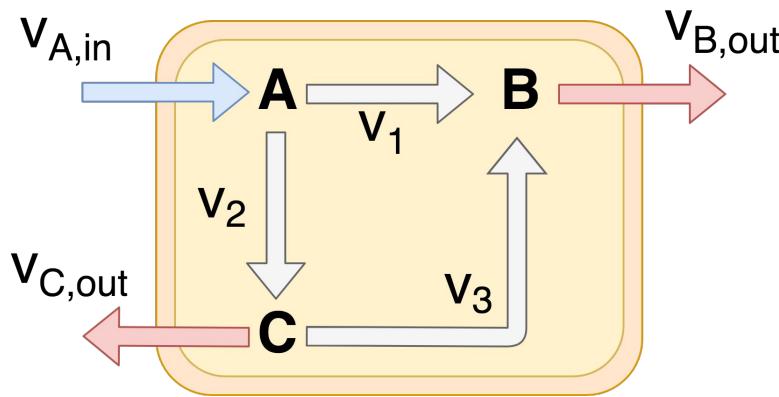
$$v_{A,in} = v_1 + v_2$$

$$v_{B,out} = v_1 + v_3$$

$$v_{C,out} = v_2 - v_3$$



Example



Constraints:

$$0 \leq v_1 \leq \infty \quad 0 \leq v_{A,in} \leq 1$$

$$0 \leq v_2 \leq \infty \quad 0 \leq v_{B,out} \leq 1$$

$$0 \leq v_3 \leq \infty \quad v_{C,out} = 0$$

Objective:

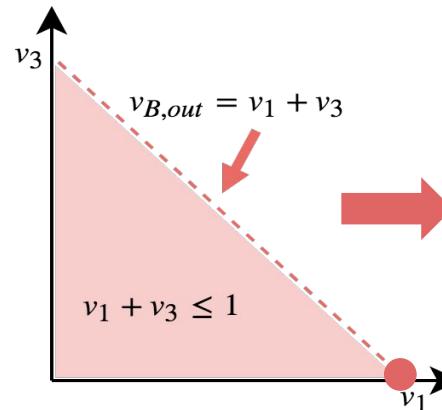
Maximize $v_{B,out}$

Steady State Assumption:

$$v_{A,in} = v_1 + v_2$$

$$v_{B,out} = v_1 + v_3$$

$$v_{C,out} = v_2 - v_3$$



Minimal Total Flux

Minimize
 $v_1 + v_2 + v_3$
 $+ v_{a,in} + v_{b,out} + v_{c,out}$

In this exercise:

1. Learn
2. Estimate the importance of each reaction for growth in different conditions
3. Compare flux distribution during day and night
4. Perform gene essentiality analysis (gene knock-outs)
- 5.
6. What else???
7. Ida from the audience?

In this exercise:

1. Use small model of the E. coli core metabolism
2. Basic steps to inspect, build, load a metabolic model
3. Perform flux balance analysis and investigate the effect of different objectives
4. Touch on alternative solutions and the variability of fluxes
5. Investigate the composition of the growth media on the growth rate
6. Perform gene essentiality analysis (gene knock-outs)

Working Environment



Install About Us Community Documentation NBViewer JupyterHub Widgets Blog

Project Jupyter exists to develop open-source software, open-standards, and services for interactive computing across dozens of programming languages.

The Jupyter Notebook interface is shown, featuring a Lorenz system visualization with three interlocking trajectories. The notebook cells contain code and text describing the system.

The Jupyter Notebook

The Jupyter Notebook is an open-source web application that allows you to create and share documents that contain live code, equations, visualizations and narrative text. Uses include: data cleaning and transformation, numerical simulation, statistical modeling, data visualization, machine learning, and much more.

Try it in your browser

Install the Notebook



Language of choice

The Notebook has support for over 40 programming languages, including Python, R, Julia, and Scala.



Share notebooks

Notebooks can be shared with others using email, Dropbox, GitHub and the Jupyter Notebook Viewer.



Interactive output

Your code can produce rich, interactive output: HTML, images, videos, LaTeX, and custom MIME types.



Big data integration

Leverage big data tools, such as Apache Spark, from Python, R and Scala. Explore that same data with pandas, scikit-learn, ggplot2, TensorFlow.

Jupyter Notebooks

<https://jupyter.org>

The Google Colaboratory interface is shown, featuring a video titled "Intro to Google Colab" with a thumbnail of a smiling man and a "Coding TensorFlow" section below it.

Google Colaboratory

<https://colab.research.google.com/>

Working Environment

Menu

Home Releases Packages Publications Docs



0 Star 182 contributors release v0.14.2

cobrapy is a python package that provides a simple interface to metabolic constraint-based reconstruction and analysis.

```
>>> import cobra
>>> model = cobra.io.read_sbml_model('Ec_core_flux1.xml')
>>> model.metabolites[3]
[<Metabolite 13pg_c at 0x112b2d160>,
 <Metabolite 2pg_c at 0x1024eb048>,
 <Metabolite 3pg_c at 0x112b2d748>]
```

The package includes simple, object-oriented interfaces for model construction (including reading to/from sbml, matlab, and json formats) and implements commonly used COBRA methods such as flux balance analysis, flux variability analysis, and gene deletion analyses.

```
>>> model.optimize()
<Solution 0.86 at 0x11272c2b0>
>>> model.summary()
IN FLUXES      OUT FLUXES      OBJECTIVES
```

A screenshot of the Cobrapy GitHub page. It features the Cobrapy logo at the top left, followed by a navigation bar with links to Home, Releases, Packages, Publications, and Docs. Below the navigation is a search bar labeled "Search docs". The main content area shows a sidebar with "1. Getting Started" and a list of 15 numbered items from "1.1. Loading a model and inspecting it" to "15. Sphinx AutoAPI Index". A code snippet in a code block titled "In [1]" shows how to import the package and load a model. At the bottom right of the page is a "Read the Docs" button.

Docs > 1. Getting Started

Edit on GitHub

1. Getting Started

1.1. Loading a model and inspecting it

To begin with, cobrapy comes with bundled models for *Salmonella* and *E. coli*, as well as a "textbook" model of *E. coli* core metabolism. To load a test model, type

```
In [1]: from __future__ import print_function
import cobra
import cobra.test

# "ecoli" and "salmonella" are also valid arguments
model = cobra.test.create_test_model("textbook")
```

The reactions, metabolites, and genes attributes of the cobrapy model are a special type of list called a `cobra.DictList`, and each one is made up of `cobra.Reaction`, `cobra.Metabolite` and `cobra.Gene` objects respectively.

```
In [2]: print(len(model.reactions))
print(len(model.metabolites))
print(len(model.genes))

95
72
137
```

When using `Jupyter notebook` this type of information is rendered as a table.

```
In [3]: model
Out[3]: Name          e_coli_core
         Memory address 0x01110ea9e8
         Number of metabolites 72
         Number of reactions 95
         Objective expression -1.0*Biomass_Ecoli_core_reverse_2cdba + 1.0*Biomass_Ecoli_core
         Compartments cytosol, extracellular
```

Just like a regular list, objects in the `DictList` can be retrieved by index. For example, to get the 30th reaction in the model (at index 29 because of `0-indexing`):

Cobrapy of OpenCobra Project

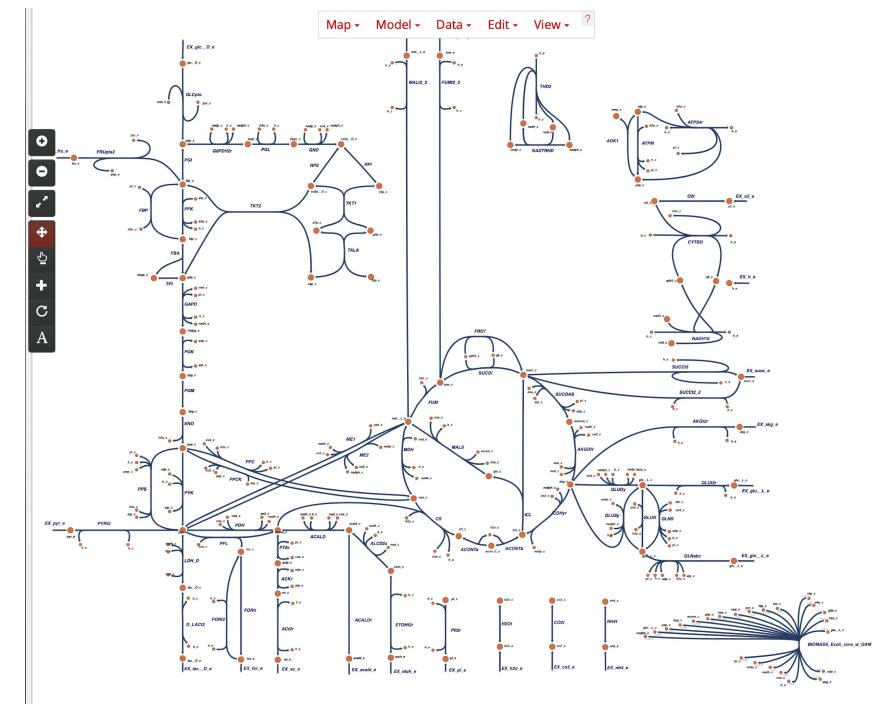
<https://opencobra.github.io/cobrapy/>

Cobrapy Tutorial & Documentation

<https://cobrapy.readthedocs.io/en/latest/>

Working Environment

The screenshot shows the Escher web application interface. At the top right is a red button labeled "What's new?". Below it is the Escher logo with a stylized brain icon. A sub-header reads "Build, share, and embed visualizations of biological pathways." A "Filter by organism" dropdown is set to "All". Below it are three tabs: "Map" (selected), "Model (Optional)", and "Tool". The "Map" tab has dropdowns for "Core metabolism (e_coli_core)" and "e_coli_core". To the right is a "Builder" dropdown. Under "Options", there are two checkboxes: "Scroll to zoom (instead of scroll to pan)" and "Never ask before reloading", both of which are unchecked. A large "Load map" button is centered below these options. At the bottom left is a section titled "Demos" with two examples: "Knockout" and "Structures". The "Knockout" demo shows a circular metabolic pathway with a blue highlighted segment and a red "Click a reaction to knock it out. Go!" button. The "Structures" demo shows a detailed chemical structure diagram for TKT1 and TALA.



Escher Maps

<https://escher.github.io>

Let's start!

1. Go to
<https://colab.research.google.com/>
2. Choose “GITHUB” and enter
“ma-blaetke”
3. Choose the repository
“2019-mb-metabolic-modelling-course”
4. Select the first notebook
“01_getting_started.ipynb”
5. Relax and follow along...

